

Welcome

INTRODUCTION TO MONGODB IN PYTHON



Donny Winston
Instructor

JavaScript Object Notation (JSON)

- objects: {}, { string : value }, { string1 : value1 ,...}
- arrays: [], [value], [value1 ,...]
- values: string , number , true , false , null , object , array

JSON ? Python

- objects: `{}, { string : value }, { string1 : value1 , ... }` ---> *dictionaries* (with `str` keys)
- arrays: `[], [value], [value1 , ...]` ---> *lists*
- values: `string , number , true , false , null , object , array` ---> `str, int/float, True, False, None, dict, list`

JSON ? Python ? MongoDB

- objects: `{}, { string : value }, { string1 : value1 , ... }` ---> *dictionaries* (with `str` keys) ---> *databases, documents, subdocuments*
- arrays: `[], [value], [value1 , ...]` ---> *lists* ---> *collections* (of documents), *arrays* (within documents)
- values: `string , number , true , false , null , object , array` ---> `str, int/float, True, False, None, dict, list, <datetime>, <re pattern>, ...` ---> `string, int/long/double, true, false, null, object, array, <date>, <regex>, ...`

The Nobel Prize API data(base)

```
import requests
from pymongo import MongoClient
```

```
# Client connects to "localhost" by default
client = MongoClient()
# Create local "nobel" database on the fly
db = client["nobel"]
```

```
# API documented at https://nobelprize.readme.io/docs/prize
for collection_name in ["prizes", "laureates"]:
    singular = collection_name[:-1]
    response = requests.get(
        "http://api.nobelprize.org/v1/{}.json".format(singular))
    documents = response.json()[collection_name]
    # Create collections on the fly
    db[collection_name].insert_many(documents)
```

Counting Documents, and Finding One to Inspect

```
# You can also access dbs and collections as attributes
```

```
assert client.nobel == db
```

```
assert db.prizes == db["prizes"]
```

```
# Count documents
```

```
n_prizes = db.prizes.count_documents({})
```

```
n_laureates = db.laureates.count_documents({})
```

```
# Find one document to inspect
```

```
doc = db.prizes.find_one({})
```

Let's practice!

INTRODUCTION TO MONGODB IN PYTHON

Finding Documents

INTRODUCTION TO MONGODB IN PYTHON



Donny Winston
Instructor

An example "laureates" document

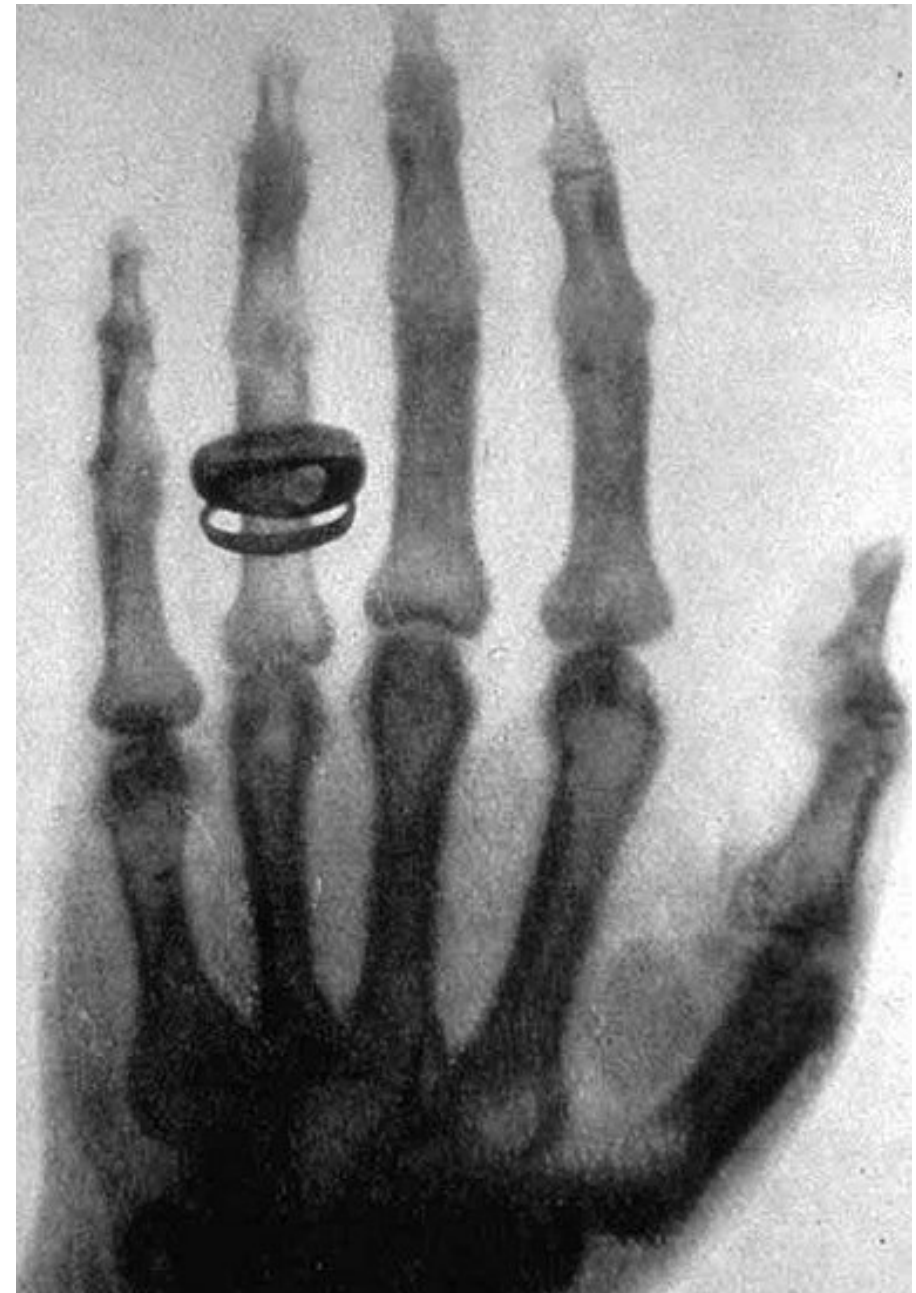
```
{ '_id': ObjectId('5b9ac94ff35b63cf5231ccb1'),  
  'born': '1845-03-27',  
  'bornCity': 'Lennep (now Remscheid)',  
  'bornCountry': 'Prussia (now Germany)',  
  'bornCountryCode': 'DE',  
  'died': '1923-02-10',  
  'diedCity': 'Munich',  
  'diedCountry': 'Germany',  
  'diedCountryCode': 'DE',  
  'firstname': 'Wilhelm Conrad',  
  'gender': 'male',  
  'id': '1',  
  'prizes': [{ 'affiliations': [{ 'city': 'Munich',  
                                'country': 'Germany',  
                                'name': 'Munich University' }],  
              'category': 'physics',  
              'motivation': '"in recognition of the extraordinary services '  
                           'he has rendered by the discovery of the '  
                           'remarkable rays subsequently named after him"',  
              'share': '1',  
              'year': '1901' } ] },  
  'surname': 'Röntgen' }
```

Filters as (sub)documents

Count documents by providing a filter document to match.

```
db.laureates.count_documents({  
    'born' : '1845-03-27',  
    'diedCountry' : 'Germany',  
    'gender' : 'male',  
    'surname' : 'Röntgen'  
})
```

1



Jimi



Amelia



Charlie



Wally



Levi



Composing Filters

```
db.laureates.count_documents({'gender': 'female'})
```

48

```
db.laureates.count_documents({'diedCountry': 'France'})
```

50

```
db.laureates.count_documents({'bornCity': 'Warsaw'})
```

2

```
db.laureates.count_documents({  
    'gender': 'female',  
    'diedCountry': 'France',  
    'bornCity': 'Warsaw'})
```

1



Query Operators

First Name

Country of Residence

Value in a Range

0 10

Submit

```
db.laureates.count_documents({  
  'diedCountry': {  
    '$in': ['France', 'USA']  
  }})
```

258

```
db.laureates.count_documents({  
  'diedCountry': {  
    '$ne': 'France'  
  }})
```

872

```
db.laureates.count_documents({  
  'diedCountry': {  
    '$gt': 'Belgium',  
    '$lte': 'USA'  
  }})
```

453

Let's Practice!

INTRODUCTION TO MONGODB IN PYTHON

Arrays, sub-documents, and dot notation

INTRODUCTION TO MONGODB IN PYTHON



Donny Winston
Instructor

A functional density

```
db.laureates.find_one({  
    "firstname": "Walter",  
    "surname": "Kohn"})
```

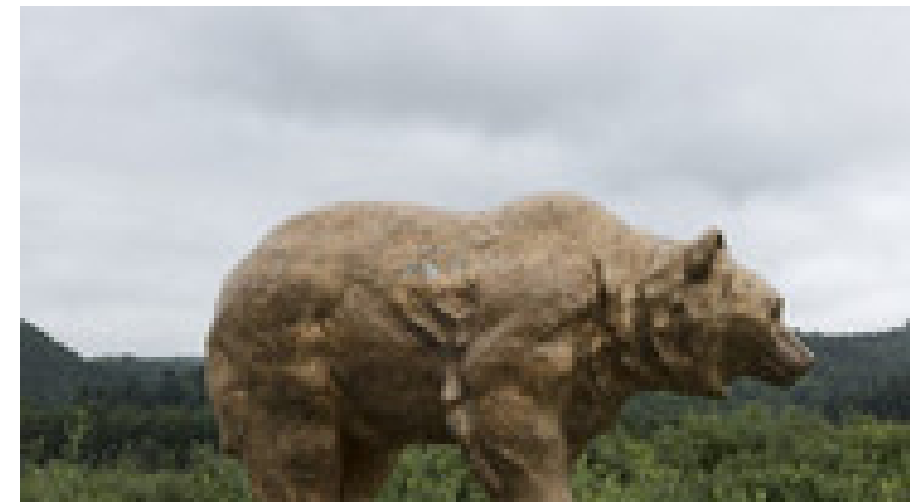
```
{'born': '1923-03-09',  
 'bornCity': 'Vienna',  
 'bornCountry': 'Austria',  
 'firstname': 'Walter',  
 'prizes': [  
     {'affiliations': [  
         {'city': 'Santa Barbara, CA',  
          'country': 'USA',  
          'name': ('University of '  
                  'California')  
         }],  
     'category': 'chemistry',  
     'motivation': (  
         '"for his development of the '  
         'density-functional theory"',  
     'share': '2',  
     'year': '1998'  
     ]],  
 'surname': 'Kohn',  
 ...} # showing partial document
```

```
db.laureates.count_documents({  
    "prizes.affiliations.name": (  
        "University of California")})
```

34

```
db.laureates.count_documents({  
    "prizes.affiliations.city": (  
        "Berkeley, CA")})
```

19



No Country for Naipaul

```
db.laureates.find_one({'surname': 'Naipaul'})
```

```
{'_id': ObjectId('5b9ec791f35b63093c3d98b7'),  
  'born': '1932-08-17',  
  'died': '2018-08-11',  
  'diedCity': 'London',  
  'diedCountry': 'United Kingdom',  
  'diedCountryCode': 'GB',  
  'firstname': 'Sir Vidiadhar Surajprasad',  
  'gender': 'male',  
  'id': '747',  
  'prizes': [{'affiliations': [],  
              'category': 'literature',  
              'motivation': ('"for having united perceptive narrative and '  
                           'incorruptible scrutiny in works that compel us '  
                           'to see the presence of suppressed histories"')},  
             {'share': '1',  
               'year': '2001'}],  
  'surname': 'Naipaul'}
```

```
db.laureates.count_documents({"bornCountry": {"$exists": False}})
```

31

31

Multiple prizes

```
db.laureates.count_documents({})
```

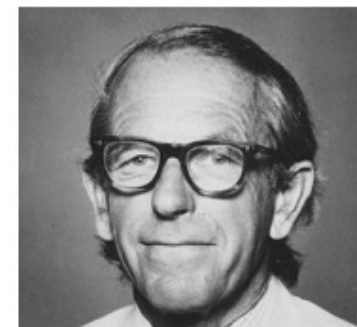
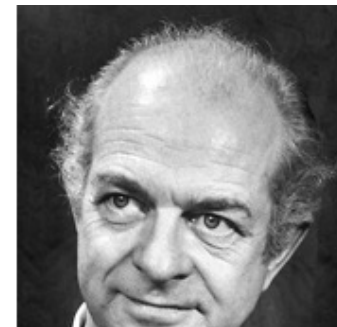
922

```
db.laureates.count_documents({"prizes": {"$exists": True}})
```

922

```
db.laureates.count_documents({"prizes.0": {"$exists": True}})
```

922



On to exercises!

INTRODUCTION TO MONGODB IN PYTHON